

## Uwiduhaye solange

### Assignment of Data structures and Logarithms

A. how does this show the LIFO natural of stacks?

So first of all stack; is linear data structure that follow LIFO (last first out) principle the element inserted, is the first to be popped out. It means that both insertion and deletion operations happens at one end only

Example

#### A: Basic

Qn1; how does this show the LIFO natural of stacks?

- In the MTNMOMO APPS when I enter payment detail step by step the last step can have failed so you can back due to insert 0 the first one removed when processing back.

Qn2; why in this action similar to popping from a stack

- In urn canvas processing back undoes the most recent action (top of stack) similar pop removes the last inserted item from the stack.

- B; application

Qn3; how could a stack enable the undo function when correcting mistake

So each transaction is pushed into a stack, occur the system pop the last transaction to undo it restoring the state that action.

Q n4; how can stack ensure forms are correctly balanced?


- In crambo forms when an opening folder is created, it is pushed into the stacks
- If every closing matches an opening the form is balanced otherwise it is invalid

C; logical

Qn5; which task is next (**top of stacks**)

- Push (CBE notes) Push ("CBE notes") → [CBE notes]
- Push ("Math revision") → [CBE notes, Math revision]
- Push("Debate") → [CBE notes, Math revision, Debate]
- Pop () → removes "Debate" → [CBE notes, Math revision]
- Push ("Group assignment") → [CBE notes, Math revision, Group assignment]

→ Top of stack = "Group assignment"

Q 6; which answer  remain in the stack after undoing

After students undoes 3 action(3pops) the last3action are removed from the stack  
; the earlier answer remains intact; for example, if stack= [q1, q2, q3, q4, q5]and 3pop are done

The [q1, q2] remain

### Advance thinking

Q7; How Does Stack This Process?

- Each step in Rwanda air booking form is pushed to the stack
- Qn8 show how stack algorithm reverses the proverb

❓ Push("umwana"), Push(ni "), Push(umutware") → [umwana numutware]

❓ Pop all → "umwana numutware"

Qn9; why does a stack suit this case better than queue

- Stack help to store the path so you can pop back step. Que would instead explore Level by level (breadth first search) which is not suitable for DFS

**Q 10; suggest a features using stack for transaction navigation**

- REDO/UNDO navigation in Bk mobile app

Each new transaction view is pushed into the stack

Going back pop the last view

Forward button pushes it again

## **PART II; QUEUE**

### **A. Basics**

Qn1; how does this show FIFO BEHAVIOUR

In restaurant Kigali, customer enter and wait in service line, the customer who arrives first is one who gets served first while the person who come last must wait until all those before them have been served.

---

### **b. Application**

**qn3; How Is This a Real-Life Queue?**

In Rwanda revenue authority offices tax payer line up in order of arrival. First Tobe served just like enqueuer (join line) and dequeuer (leave after services)

---

### **c. LOGICAL**

**qn5: who is at the front now?**

- Start: Enqueuer("Alice") → [Alice]
- Enqueuer("Eric") → [Alice, Eric]
- Enqueuer("Chantal") → [Alice, Eric, Chantal]
- Dequeuer () → Alice leaves → [Eric, Chantal]
- Enqueuer("Jean") → [Eric, Chantal, Jean]

Eric is at the front now.

Q6. Explain how a Queue ensures fairness.

RSSB, pension applications are handled in **arrival order**. This FIFO rule makes sure nobody skips ahead, so everyone is treated equally and fairly.

#### d. ADVANCED THINKING

---

Q7: Explain how each maps to real Rwandan life.

- **linear queue** = At a wedding buffet, people line up, and once served, they leave.
- **circular queue** = Buses at Nyabugogo rotate; after finishing one trip, they rejoin the end of the line for the next turn.
- **deque** = boarding a bus, where passengers can sometimes enter from the front door or rear door.

Q8: How can queues model this process?

☞ At a Kigali restaurant, food orders are **enqueue** when placed. When the chef finishes preparing, the order is **dequeue** and customer is called, the model order-taking and serving

Q9; why is This a priority queue, not a normal queue?

At CHUK hospital, critical patients (emergencies) are treated first, even if they arrived later.

This breaks the normal FIFO rule, so it is a priority queue

Q10; how would queue fairly match drivers and student?

In moto/e-bike app, driver's **enqueue** when available, and student **enqueue** when requesting rides

The system **dequeues** in order, pairing the first available driver with the first waiting student,

Ensuring fairness queues are not just abstract data structures in computer science

Conclusion; but also real systems we experience every day in Rwanda — from restaurants and service centers to hospitals and transport. They work on the principle of **FIFO (First-In-First-Out)**, ensuring fairness, order, and efficiency. In special cases like hospitals, **priority queues** are used to save lives. Understanding queues helps us appreciate how computing models are applied in solving everyday real-world problems.